

FIG. 1

```

00. procedure ProcessDistVect(et, m, d, j)
01. { et is the type, m is the neighbor, d is the distance, j is the destination }
02. begin
03.   if (j = thisNode  $\wedge$  et = QUERY) then send [REPLY, j, 0]; endif
04.    $D_{jm}^i \leftarrow d$ ;
05.    $D_j^i \leftarrow \min\{D_{jk}^i + l_j^i | k \in N^i\}$ ;
06.    $SD_j^i \leftarrow \min\{D_{jk}^i + l_k^i | k \in S_j^i\}$ ;
07.   if (stateji = PASSIVE  $\vee$  stateji = ACTIVE  $\wedge$  last reply is received for j) then
08.      $FD_j^i \leftarrow \min\{D_j^i, RD_j^i\}$ ;
09.     if ( $D_j^i > RD_j^i$ ) then
10.       stateji  $\leftarrow$  ACTIVE;
11.       if (et = QUERY) then
12.          $QS_{jm}^i \leftarrow m$ ;
13.       endif
14.        $RD_j^i \leftarrow SD_j^i$ ;
15.        $\forall k \in N^i$ , send [QUERY, j,  $RD_j^i$ ] to neighbor k;
16.     else
17.       stateji  $\leftarrow$  PASSIVE;
18.       foreach k  $\in N^i$  do
19.         if ( $k \in QS_j^i \vee (k = m \wedge et = QUERY)$ ) then send [REPLY, j,  $D_j^i$ ];
20.         else if ( $RD_j^i \neq D_j^i$ ) send [UPDATE, j,  $RD_j^i$ ];
21.       endif
22.     done
23.      $RD_j^i \leftarrow D_j^i$ ;
24.      $QS_j^i \leftarrow \emptyset$ ;
25.   endif
26.   else
27.     if (et = QUERY) then
28.       if ( $m \in S_j^i \wedge SD_j^i > RD_j^i$ ) then  $QS_j^i \leftarrow QS_j^i \cup m$ ;
29.       else send [REPLY, j,  $RD_j^i$ ];
30.     endif
31.   endif
32. endif
33.  $S_j^i \leftarrow \{k | D_{jk}^i < FD_j^i\}$ ;
34. end

```

FIG. 2

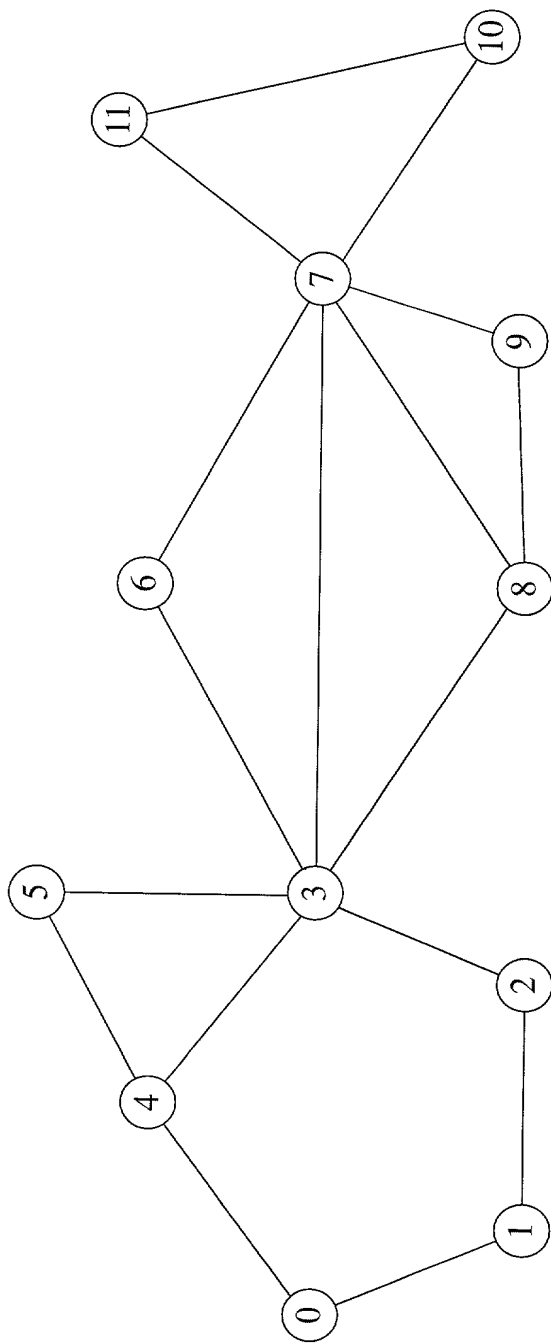


FIG. 3

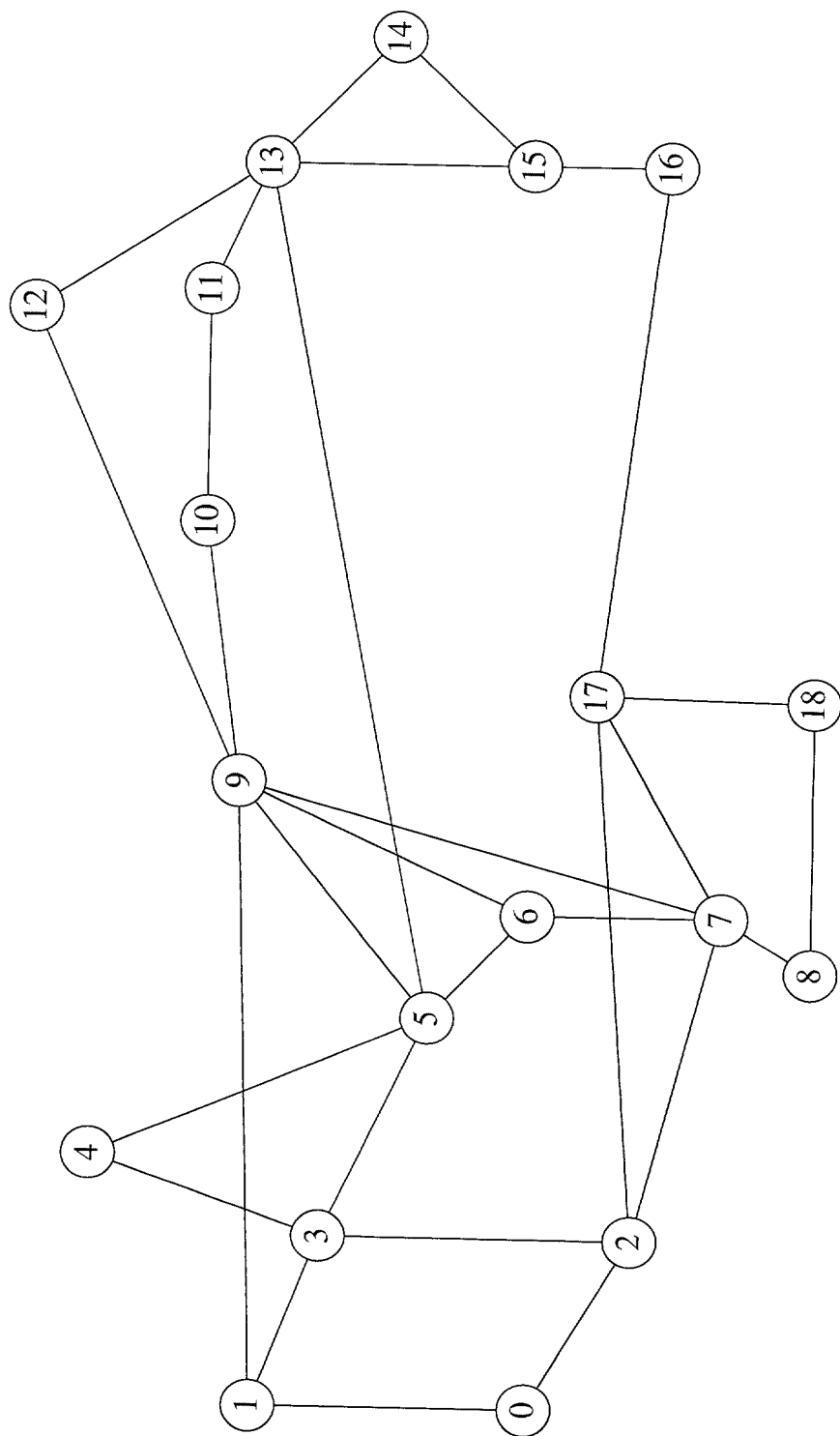


FIG. 4